INFORMATION CREDIBILITY ASSESSMENT AND META DATA MODELING IN INTEGRATING HETEROGENEOUS DATA SOURCES
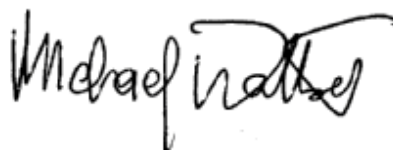
Louisiana State University

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

**AIR FORCE RESEARCH LABORATORY
INFORMATION DIRECTORATE
ROME RESEARCH SITE
ROME, NEW YORK**

This report has been reviewed by the Air Force Research Laboratory, Information Directorate, Public Affairs Office (IFOIPA) and is releasable to the National Technical Information Service (NTIS). At NTIS it will be releasable to the general public, including foreign nations.

AFRL-IF-RS-TR-2002-298 has been reviewed and is approved for publication.

APPROVED:

RAYMOND A. LIUZZI
Project Engineer

FOR THE DIRECTOR:

MICHAEL L. TALBERT, Maj., USAF
Technical Advisor, Information Technology Division
Information Directorate

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 074-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE NOVEMBER 2002	3. REPORT TYPE AND DATES COVERED Final Jul 00 – Jan 02	

4. TITLE AND SUBTITLE
INFORMATION CREDIBILITY ASSESSMENT AND META DATA MODELING IN INTEGRATING HETEROGENEOUS DATA SOURCES

6. AUTHOR(S)
Peter P. Chen

5. FUNDING NUMBERS
C - F30602-00-2-0605
PE - 62232N, 62702F
PR - R427
TA - 02
WU - P3

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Louisiana State University
298 Coates Hall
Baton Rouge Louisiana 70803

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Research Laboratory/IFTB
525 Brooks Road
Rome New York 13441-4505

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

AFRL-IF-RS-TR-2002-298

11. SUPPLEMENTARY NOTES

AFRL Project Engineer: Raymond A. Liuzzi/IFTB/(315) 330-3577/ Raymond.Liuzzi@rl.af.mil

12a. DISTRIBUTION / AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 Words)*
This effort identified several key issues and proposes a framework to solve the problem of credibility and validity assessment of information from various data sources for information fusion. This report outlines some of the important steps in a framework for an information validity assessment. The report also describes some algorithms for conflict resolution. The effort also proposes a prototype of a decision-support system to support the estimation of the composite data values from heterogeneous databases with different validity assessment values. As part of this effort this prototype has been implemented using Java and Oracle DBMS version 8i for helping people to make decisions under conflicting data situations and for information validity assessment based on the proposed meta-data conceptual modeling methodology.

Finally, several issues closely related to information credibility assessment such as meta data modeling and reverse-engineering of existing database schemas into conceptual models are examined.

14. SUBJECT TERMS
Database Management, Information Credibility, Information Validity, Data Credibility, Information Fusion, Data Quality, Heterogeneous Data, Data Integration, Database Schema, Decision-Support Systems

15. NUMBER OF PAGES
37

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UL

TABLE OF CONTENTS

LIST OF FIGURES

Acknowledgment

1 Introduction

Information is viewed as an active interaction of knowledge and data. An active paradigm for information management is proposed for investigation of the proposed information services. Under this active paradigm, information services are viewed as behaviors of the information system to be implemented as "intelligent" shareable software modules to provide cooperation among users, applications/agents, and the information system environment.

Intelligent extraction and validation of information from multiple database systems that are of different models (e.g. relational, Object-Oriented, flat files, etc.) are the key prerequisites to efficient knowledge integration for decision support. The extraction and validation capabilities can serve as mediators between the users or applications/agents and the underlying information system in carrying out complex tasks.

Current commercial database systems (e.g. Sybase, Oracle, etc.) and most of the research prototypes focus on providing information for the users through query mechanisms. Our effort attempts to provide an active information management framework, not only allowing applications or users query or extract information from databases of different model (e.g. relational, Object-Oriented, flat files etc.), the active information system cooperates with the users by automatically informing the applications or users of the necessary information whenever critical situations occur. One of the key aspects in our effort is to develop a framework and techniques for information credibility assessment.

In many situations (military/civil environments and our daily lives), we encounter the information credibility problem in many different forms: the data from different sources are supposed to be the same, but actually, they are different. Here are some examples:

- Three sensors are tracking the same military target, but each is reporting a different location of the target.
- The intelligence reports provide conflicting data. For example, at the end of the World War II, Hitler received conflicting intelligence reports on the location and date of the pending invasion of the European Continent by the Allied Forces.
- Two databases in different locations started with the same data and schema. After a while, these two copies are drifting apart because new data fields are added (and some existing data fields are deleted or changed) for the schema of each database due to local needs. In addition, some of the data in the unchanged data fields could also be different because of maintenance and other errors.
- Some of the employee data (of the same employee) in different databases of the same organization are different.

How can we handle this problem? We think there are at least three things that will be very useful:

- A framework for analysis
- Computational formulae or algorithms for conflict resolution.
- A software program that help people make decisions.

We will describe each of these three subjects in the following sections. In addition, we will discuss other relevant issues such as meta data modeling and reverse engineering of existing database schemas. In terms of meta data, we will present a conceptual model of meta data that also includes the interactions of Event_Condition_Action (ECA) rules [Chak95].

We have done a survey of current literature on meta-data conceptual modeling of information credibility assessment. Even though there are quite a few researchers working on different aspects of meta-data research (see, for example, [IEEE99], there are very few working on meta-data conceptual modeling of information credibility assessment. We can identify about three research work somewhat relevant to what we propose to do in this project:

(1) The Meta-database project by Chen Hsu at RPI [Hsu]. Integration has become a self-evident goal in today's manufacturing enterprises. Among the barriers are major gaps in information technology regarding multiple systems operating concurrently over different geographical regions. Dr. Hsu discusses a meta-data approach to the integration problem using a two-level Entity-Relationship (ER) model. His two-level ER model is somewhat similar to what we are proposing. However, his emphasis is more on manufacturing system integration while our emphasis is more on data integration for data from various sources.

(2) The ER modeling of data quality work done by Veda Storey and Richard Wang [StWa98]: They used ER modeling to describe the data quality of production data in the databases. While their work is interesting and very relevant to what we are doing, our approach to ER modeling of meta-data is different from what they did. They basically modeled both the meta-data and production data in the same level of ER diagrams while we have been and will use two-level ER diagrams to separate the modeling of production data and meta-data. We believe that our approach is cleaner and less error-prone, and therefore, can produce a software system that is more easily verifiable.

(3) The ER modeling of metadata in the European Data Warehousing Project [ArFr99, CDLN99, JaVa97, JeQJ98, StJa96]: A multi-country project was funded by European Commission to study the best way to design and to organize data warehouses. Meta-data approach using the ER modeling was proposed and investigated by several researchers in the project. While their approaches are relevant and interesting, our approach concentrates on information credibility assessment than on data warehouse design.

2   A Preliminary Framework for Information Credibility Assessment

We are proposing a framework for information credibility assessment with the following steps:
1. Identification of relevant factors in your environment that has impacts on the "information credibility." Out of all the factors influencing "information credibility," we believe "information validity" is a dominant factor.
2. "Information validity" has at least three components: "reliability," "freshness," and "believability."
3. Each data element can be associated with an "Information Validity (InfV) assessment value (to be discussed in Section 2.2).

4. If the InfV values are only available to individual data elements, computational formulae are needed to calculate the composite InfV values for the aggregate objects in the higher levels (for example, the InfV values for a particular row in a database table, or the InfV value for a particular database table, or the InfV value for a particular database).

5. The InfV values can be modeled in a meta-data conceptual model and be kept in a meta-data database.

6. In existing databases, the conceptual schemas (models) may not be documented. Reverse engineering tools may be used to re-construct the conceptual schema from existing data structures.

We will discuss these steps in details in the next few sections.

2.1 Definitions of Basic Terminology

Many of the terms in this field are very confusing because different people have used them in different ways. We think it will be very useful to the community if there is a common definition of some basic terms. Here, we are making an attempt to give some definitions of several important terms:

- **Reliability**: It concerns with the impacts and implications of the possible malfunctions of the supporting platforms (such as hardware, operating systems, file and database systems).
- **Believability**: It concerns with how much you can trust data sources (person, sensor, etc.) and data transmission channels.
- **Freshness**: It concerns with the age of data. The data values usually changes with time. Therefore, the data has a higher chance to be correct if it was supplied/input/given very recently.
- **Quality**: It concerns with the percentage of deviation (or non-deviation) from the best or the original form of the data.
- **Credibility**: It is a composite measure of many factors about the same data.
- **Consistency**: It measures whether a piece of data contradicts with another piece of data. For example, a person's age may contradict with his birth date.
- **Integrity**: It measures whether the data has been maliciously modified. It is possible that data are consistent but the data integrity is not maintained. For example, if someone's age and birth date were changed to incorrect values but still consistent with each other, the data integrity is not maintained.

In Figure 2.1-1, we organize the concepts into a tree structure. "Credibility" is a composite measure of various concepts. The main factor that influences "credibility" is "validity," but other factors (such as "quality," "integrity," "consistency," and "security") also have impacts on it. In this report, we will not discuss in detail the inter-relationships among these four concepts mentioned above, and we will leave this topic as one of the future research topics.

Although our definitions and the concept tree in Figure 2.1-1 may not be agreeable with some other researchers, we believe that we have made some important contributions because there are very few such attempts by others to clarify these terms and concepts. In the future, we will continue to modify and improve these definitions.
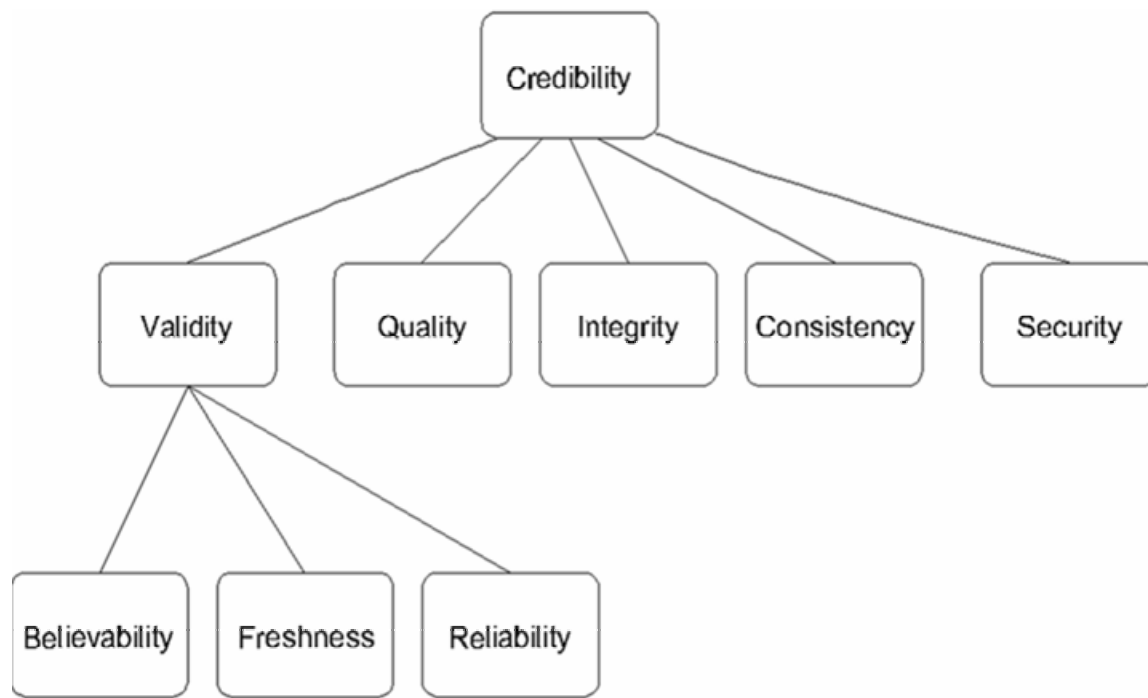
Figure 2.1-1. Concept Tree

2.2.   Factors of Information Validity Assessment Value

As discussed in the previous section, "validity" is a major component of "credibility." Because we have not studied several other components of "credibility" such as "quality," "integrity," "consistency," and "security," we now turn our attention to "validity," that can be assessed a numeric value. First, we need to know the "factors" that produces the Information validity (InfV) value.

Each data source will be given an InfV value. However, InfV is a composition of several factors including at least the following factors (see Figure 2.1-1):

- <u>Reliability of the hardware and software that the database resides</u>: Collecting the past operational data of the hardware and software malfunction frequencies can be useful in assessing this reliability value in the past time periods. The reliability value is between "0" and "1."

- <u>Freshness of the data</u>: If the data was just updated, the freshness factor could be 1.0. If the data was updated one year or more ago, the freshness could be "0". We can define a function of values between "0" and "1" for any "age" of data between "one year" and "0".

- <u>Believability of the data</u>: how believable of the data depends on where the data came from. For example, if the data comes from a very believable source, it will be certainly more believable than the data comes from an unreliable source. The believability factor has a value between "0" and "1".

3   Computational formulae or algorithms for conflict resolution

3.1   The Need for Conflict Resolution Formulae or Algorithms

If we derive 3 sets of values from three different sources for the same query, then the question is which one is correct. Or, how do we derive an InfV (Information Validity) value from the sets of values derived? One approach to solve this problem is to associate a value between 0 and 1 to the BASE data value (instead of assuming them to be always 1 as in the case of traditional databases). The InfV values can be stored either in the database or the meta database based on the original conceptual model or the conceptual model resulting from reverse engineering/modeling.

Typically, the value associated with the base data depends on the source of data acquisition, update frequency, the confidence in the system from which the data is obtained or the confidence in the instrument from which the data is acquired, etc. Once a measure is associated with the base data, it can be combined (or new values can be derived) in a number of ways.

3.2 Several Conflict Resolution Formulae or Algorithms
The following is a collection of some simpler formulae or algorithms.

1. Take the max of the values
2. Take the min of the values
3. Take the average of the values
4. Take consensus or majority vote
5. Discard values below or above a threshold and apply the above to the resulting values.
6. Arbitrarily pick one
7. Apply a function on the probability to compute the new values
8. Dynamically collecting probabilities by the system

3.2.1 A Proposed Algorithm

In the following, we are going to describe a new algorithm to derive a "composite" value.

Algorithm 1 (for two data sources):

Let V_1, V_2 be the data values of the same data element from two different sources, and Let C_1, C_2 the data validity assessment value of the two data values. Let V^*, C^* be the estimated data value and the associated data validity assessment value based on these two given data values and their associated data validity assessment value, and they can be derived by the following formula:

$$V^* = \begin{array}{ll} V_1 * (C_1/(C_1+C_2)) + V_2 * (C_1/(C_1+C_2)) & \text{if } C_1+C_2 \neq 0, \\ (V_1 + V_2)/2 & \text{if } C_1=C_2=0. \end{array}$$

$$\tag{1}$$

$$C^* = \begin{array}{ll} C_1 * (C_1/(C_1+C_2)) + C_2 * (C_1/(C_1+C_2)) & \text{if } C_1+C_2 \neq 0, \\ (C_1 + C_2)/2 & \text{if } C_1=C_2=0. \end{array}$$

$$\tag{2}$$

Example #1:
Given these values: $V_1 = 10, V_2 = 100, C_1 = 0.6, C_2 = 0.8$, the values of V^* and C^* can be derived as follow:

$$
\begin{aligned}
V^* &= V_1 * (C_1/(C_1+C_2)) + V_2 * (C_1/(C_1+C_2)) \\
&= 10 * (0.6/(0.6+0.8)) + 100 * (0.8/(0.6+0.8)) \\
&= 4.29 + 57.14 \\
&= 61.43 \\
C^* &= C_1 * (C_1/(C_1+C_2)) + C_2 * (C_1/(C_1+C_2)) \\
&= 0.6 * (0.6/(0.6+0.8)) + 0.8 * (0.8/(0.6+0.8)) \\
&= 0.2571 + 0.4571 \\
&= 0.7142
\end{aligned}
$$

Algorithm 1.1. (for "n" data sources):

Let $V_1, V_2, \ldots, V_n$ ($n > 2$) be the data values of the same data element from "n" different sources, and Let $C_1, C_2, \ldots, C_n$ the data validity assessment value of the "n" data values. Let V^*, C^* be the estimated data value and the associated data validity assessment value based on these "n" given data values and their associated data validity assessment value, and they can be derived by the following formula:

$$V^* = V_1 * (C_1/(C_1+C_2 \ldots +C_n)) + V_2 * (C_2/(C_1+C_2 \ldots +C_n))+\ldots+ V_n * (C_n/(C_1+C_2 \ldots +C_n))$$
$$\text{if } C_1+C_2+\ldots+C_n \neq 0,$$

$$= (V_1 + V_2 +\ldots + V_n)/n \qquad\qquad \text{if } C_1=C_2 \ldots =C_n = 0.$$
$$(3)$$

$$C^* = C_1 * (C_1/(C_1+C_2 \ldots +C_n)) + C_2 * (C_2/(C_1+C_2 \ldots +C_n))+\ldots+ C_n * (C_n/(C_1+C_2 \ldots +C_n))$$
$$\text{if } C_1+C_2+\ldots+C_n \neq 0,$$

$$= (C_1 + C_2 +\ldots + C_n)/n$$
$$\text{if } C_1=C_2 \ldots =C_n = 0.$$
$$(4)$$

Algorithm 2. Algorithm to Calculate the Confidence Level of the Whole Table

Here, we describe an algorithm and the rationale of the algorithm to calculate the confidence level of the whole table from the confidence level of each data element in the table. A simple version of this algorithm has been implemented.

Totally we have three sets of input:
 1. Table S: input sample data
 2. Table C: Confidence level value for each date field
 3. Two weight arrays assigned by columns and rows
Sample data:
 S

s#	sname	status
002	J Smith	30
005	F Jones	50
126	K Landry	100

C ($0<=c_i<=1$):

s#	sname	status
1	0.82	0.82

1	0.46	0.32
1	0	0.25

If table C has dimension of 3*3, the confidence level c_{ij} (i,j=1,2,3) will looks like the following:

$$c_{11} \quad c_{12} \quad c_{13}$$

$$c_{21} \quad c_{22} \quad c_{23}$$

$$c_{31} \quad c_{32} \quad c_{33}$$

Given a table with dimension 3*3, how do we calculate the confidence level data in each row or in each column. In order to do the weighting average of the confidence level, we need to assess the "weighting factor" for each row and each column of the confidence level data elements. For example, the weighting factors for a 3*3 confidence level table will look like the following:

Weight($0 <= w_i <=$ #element):

$$w_{c1} = 0.7 \qquad w_{c2} = 1.9 \qquad w_{c3} = 0.4$$

$$w_{r1} = 0.3$$

$$w_{r2} = 1.5$$

$$w_{r3} = 1.2$$

$\sum_{i=1}^{n} w_{ri} = n$ (Validity check is needed here)

$\sum_{i=1}^{m} w_{ci} = m$ (Validity check is needed here)

n is the total number of records in the database table

m is the total number of attributes in the database table

The weighted average confidence level for each column:

$$c_{c1} = (w_{r1}c_{11} + w_{r2}c_{21} + ...) / \#$$
$$= \sum_{i=1}^{n} w_{ri} * c_{i1} / \sum_{i=1}^{n} w_{ri}$$
$$= (0.3*1 + 1.5*1 + 1.2*1)/3$$
$$= 1$$

$$c_{c2} = (0.3*0.82 + 1.5*0.46 + 1.2*0)/3$$
$$= 0.312$$

. . . . . .

Note: we need to do a validity check on w values as $\sum_{i=1}^{n} w_{ri} == n$ and $\sum_{i=1}^{m} w_{ci} == m$

Now we do the similar calculation for each row:

$$c_{r1} = (w_{c1} c_{11} + w_{c2} c_{12} + ...)/\#$$
$$= \sum_{i=1}^{m} w_{ci} * c_{1i} / \sum_{i=1}^{m} w_{ci}$$
$$= (0.7*1+1.9*0.82+0.4*0.82)/3$$
$$= 0.862$$

. . . . . .
. . . . . .

Result of calculation: Two weighted confidence level arrays for the whole table

$$c_{c1} = 1 \qquad c_{c2} = .312 \qquad c_{c3} = .342$$

$c_{r1} = .862$

$c_{r2} = .765$

$c_{r3} = .267$

The total confidence level of the table then is:
By column:
$$c_{T1} = \sum_{i=1}^{m} w_{ci} * c_{ci} / \sum_{i=1}^{m} w_{ci} = (0.7*1+1.9*0.312+0.4*0.342)/3 = 0.4765$$
Here shows an example of getting different values of "c" if we reverse the order of the calculation (from column calculation to row calculation)
By row:
$$c_{T2} = \sum_{i=1}^{n} w_{ri} * c_{ri} / \sum_{i=1}^{n} w_{ri} = (0.3*0.862+1.5*0.765+1.2*0.267)/3 = 0.5755$$

3.3 The Use of Dampster & Shafer's Theory with the Proposed Algorithms

Dampster & Shafer's Theory of Evidence and its application to knowledge extraction and integration: As explained before, data coming from different sources is likely to have different levels of confidence. If the confidence or reliability of data (for example, it may be based on the precision of the instrument, confidence associated with informers, etc.) is known or stored as part of the data, then it will be possible to associate a level of confidence or reliability with data derived or extracted from the base data. The theory of Evidence deals with how to resolve conflicting evidence. From our investigation, it seems that Dampster & Shafer's theory can be extended to the "conflict resolution of data." This is a very interesting and challenging problem and takes a longer time to find a clear solution because the problem we have at hand is more complicated and may require extensions or reformulation of this problem using, for example, domain meta data or correlation of data to establish the measure of level of confidence and reformulation of Dampster & Shafer's formulae

4   Metadata Conceptual Modeling, Active Modeling, & Reverse-Engineering

4.1   Evaluation of Existing Approaches

We have studied some of the work done at MIT Data Quality Project (specifically, the Quality ER Model) and the work done at ESPRIT DWQ (Foundations of Data Warehouse Quality) project. We had studied carefully the Quality ER Model developed by Storey and Wang. They divided the data into 3 major types: the product data, the product quality data, and the data quality data.   We think this division is a useful concept but the way they put all three data types into the same "layer" of an Entity-Relationship (ER) diagram is not suitable for our needs. We think the product data and the product quality data could be put into the same "layer" of an ER diagram, but the data quality data should be treated as "meta data" and be put into a separate ER diagram.

4.2   Our Proposed Approach to the Metadata Model

We propose a meta-data conceptual methodology as follow (see Figure 4.2-1):

1. To recognize and identify the entities and relationships in the real world: for example, we can recognize cities and communication towers.
2. To represent these real world entities and relationships using ER diagrams.
3. To convert the ER diagram into table structures in the (relational) databases and to populate these tables with data: In our example, we will have "C_Tower" table, "Located_in" Table, and "City" Table. We will also fill up these tables with data.
4. To model the quality of the data in the (relational) databases: Using ER diagrams to model the meta data of the (relational) databases. In our example, each table is a meta-entity, and each data element is also a meta-entity. The "Table" meta-entity and the "Data Element" meta-entity have a meta-relationship called, "Consists_of." The "Table" meta-entity has a meta-attribute called, "Information Validity Value of Table," and the "Data Element" meta-entity has a meta-attribute called, "Information Validity Value of Data Element."

5. To convert the ER diagram in (4) into table structures in the (relational) databases and to populate these tables with data: In other words, we are building a (relational) database to hold the meta-data concerning with the Information Validity Value of the data in the tables and data elements in the production databases.

Figure 4.2-2 is an expanded view of the ER diagram in the right side of the Figure 4.2-1. Basically, we are adding another meta-entity type called, "Database," that consists of "Tables." We also have another meta-attribute called, "Information Validity Value of Database," which is an aggregate of "Information Validity Value of Table," in a particular database. Similarly, "Information Validity Value of Table" is an aggregate of "Information Validity Value of data elements," in a particular table. The Algorithm 2 in Section 3.2 is an algorithm to aggregate the Information Validity (InfV) Value of each data elements in a table into the Information Validity Value of the whole table. As discussed in Section 2.1, the Information Validity Value is a function of at least three factors: believability of the source, freshness of data, and the quality of data. In Figure 4.2-2, we indicate in the diagram

that the Information Validity Value of a data element is an aggregate of three attributes: believability of the source, freshness of data, and the quality of data. Algorithms 1.0 and 1.1 in Section 3.2 are proposed for the use in resolving the conflicting data in different databases based on the meta-data (such as InfV values) stored for the data elements in each database.

How do the two levels of ERD's related to each other? We will explain their relationships using two figures: Figure 4.2-3 and Figure 4.2-4. In Figure 4.2-3, we have three tables of data: a table of data on "communication tower" entities, a table of data on "city" entities, and a table of data on the relationships between communication tower entities and city entities. Each of these tables is assessed with a particular Information Validity value. Note that we are concerning with "instances" here (be it "entity instance," "relationship instance," or "value instance'). In Figure 4.2-4, we are concerning with "types" instead of "instances." For example, "Comm. Tower" entity type and "City" entity types in Figure 4.2-3 are instances of the "Entity Type" in the meta entity type in Figure 4.2-4. Similarly, the "Located_in" relationship type in Figure 4.2-3 is an instance of the "Related_to_1" meta relationship type. Similarly, on the right hand side of Figure 4.2-4, we are concerned with the entity types of data tables. There are two major types of data tables: one is the type of tables containing data about entities such as the table of data on "Comm. Tower" entities and the table of data on "City" entities in Figure 4.2-3. Another major type of table is the type of tables containing data about relationships such as the table of data on "Located_in" relationships in Figure 4.2-3. In Figure 4.2-4, the ERD in the left hand side is in Level 1, and the ERD in the right hand side is in Level 2.

These two ERD's are related by relationship types as shown in the middle column in Figure 4.2-4. The "Table (Data) on Entities" entity type in Level 2 and the "Entity type" meta entity type in Level 1 have a relationship type called, "Related_to_2" between them because the data in Level 2 are the description of entities in level 1. Similarly, the "Table (Data) on Relationships" entity type in Level 2 and the "Relationship type" meta entity type in Level 1 have a relationship type called, "Related_to_3" between them.
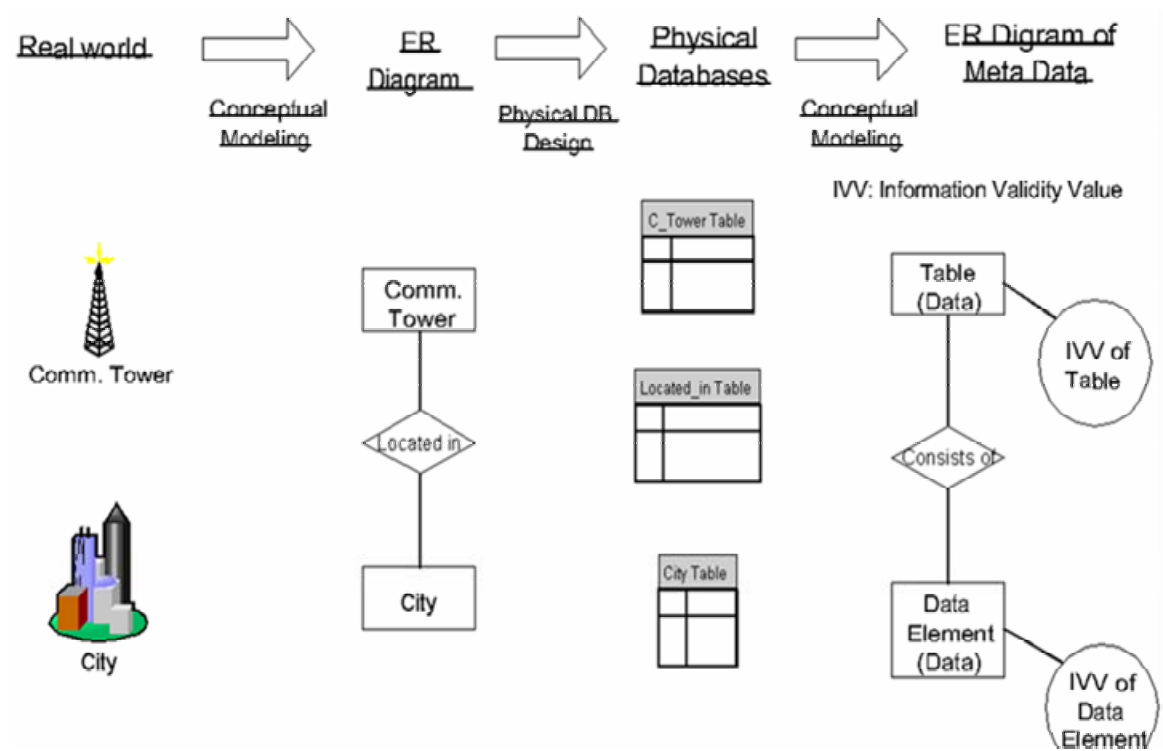
Figure 4.2-1. Overview of Meta Data Conceptual Modeling Methodology (Two-Level Modeling)

Note that we have developed the computation formula to calculate the IVV (Information Validity Value) of the whole table for the IVV's of the data delements in the table. To aggregate into an IVV of the whole database, similar formula can be used.

Figure 4.2-2. ER Diagram to represent Computation Formula for Aggregating Factors

Figure 4.2-3. Modeling Information Validity Values of 3 Tables of Data
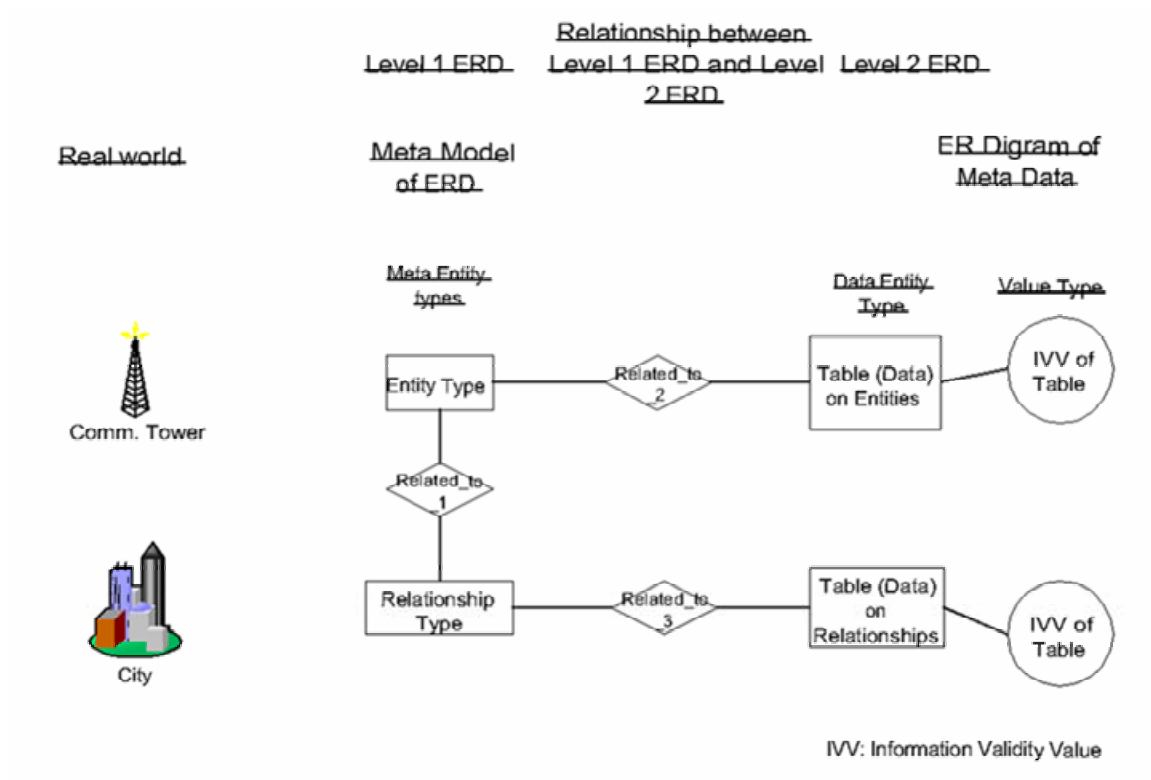
Figure 4.2-4. Relationship between two Levels of ERD's
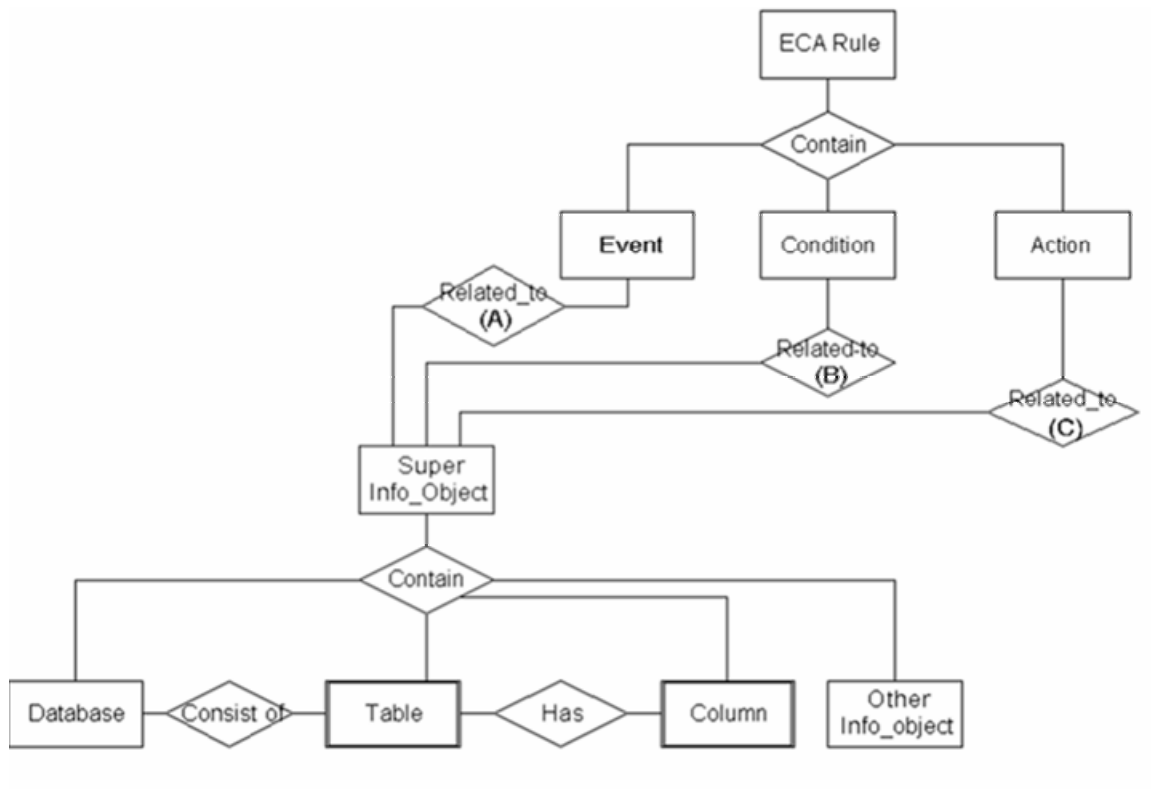
Figure 4.2-5. A Metadata Conceptual Model

How can we link this meta-data model to some research work on ECA (Event, Condition, and Action) rules of a system [Chak95]?   We propose the following "entity" types in the metadata conceptual model with the associated "attributes (See Figure 4.2-5):

- Entity Type "Database" with attributes: DB_Name, Reliability, Freshness, Believability, Data Validity Assessment (DVA) and other attributes. The DVA is a derived attribute from several attributes: Reliability, Freshness, and Believability.
- Entity Type "Table" with attributes: Table_Name, etc.
- Entity Type "Column" with attributes: Column_Name, etc.
- Entity Type "Event" with attributes: Event_ID, etc.
- Entity Type "Condition" with attributes: Condition_ID, etc.
- Entity Type "Action" with attributes: Action_Name, etc.
- Entity Type "Other_Info_Object" with attributes: Info_Obj_Name, etc.
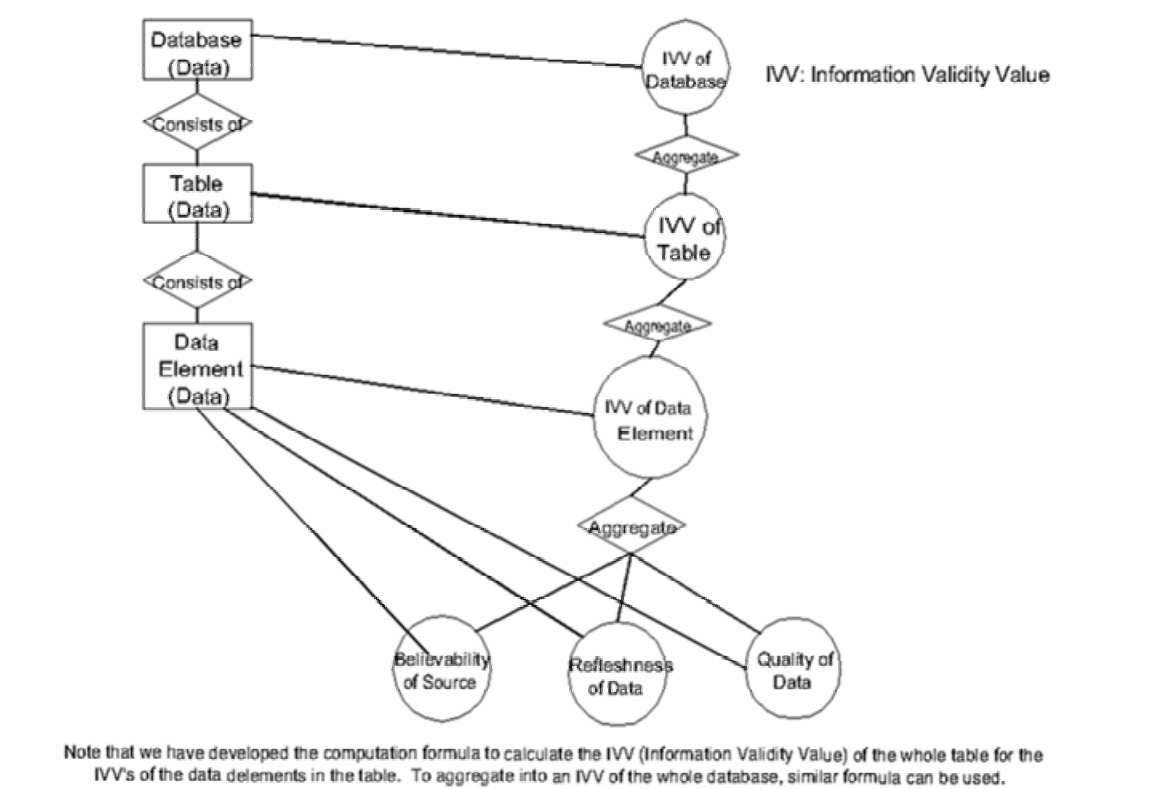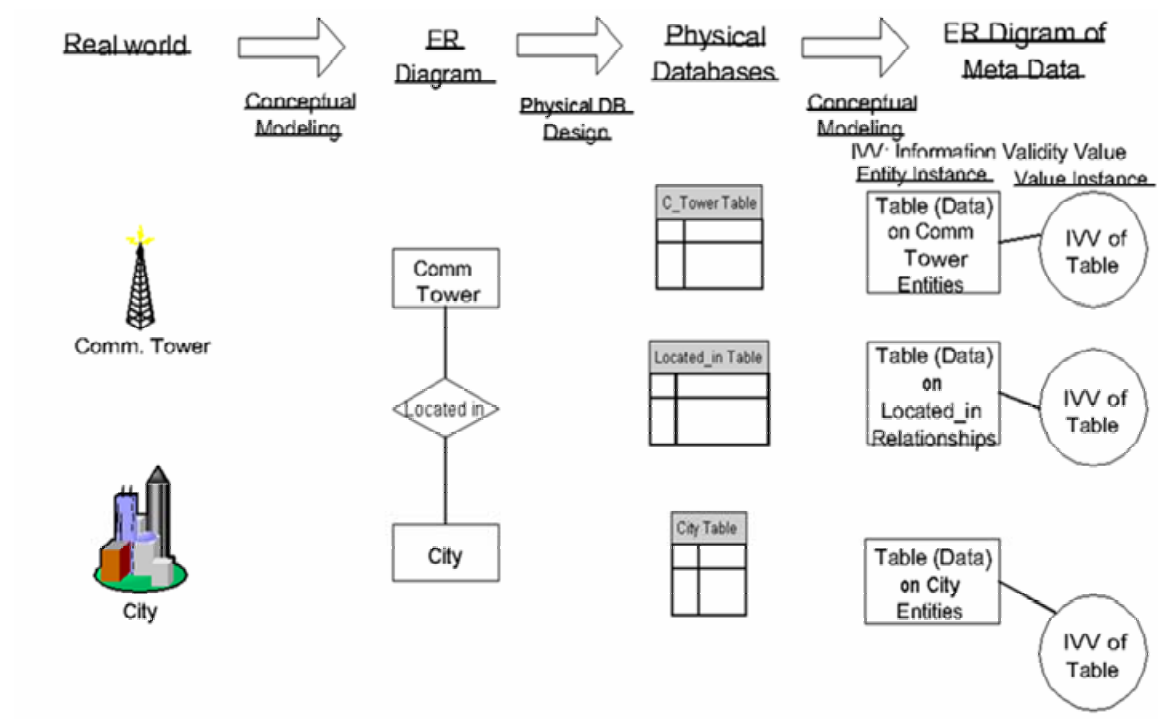- Entity Type "Super_Info_Object with attributes: Super_Info_Obj_Name, etc. This entity is an aggregation of several entity types: Database, Table, Column, and Other_Info_Object.
- Entity Type "ECA_Rule" with attributes: ECA_Rule_ID, etc. This entity is an aggregation of three entity types: Event, Condition, and Rule.
- Special Entity Type "Time" with attributes: Time_instance
   There exist several relationship types between these entity types:
- Database "consist_of" Tables: it is an ID-dependent relationship, that is, the Table needs DB_Name (in addition to Table_Name) to identify itself.
- Table "has" Columns: it is also an ID-dependent relationship, that is, the Column needs DB_Name and Table_Name (in addition to Column_Name) to identify itself.
- Event, Condition, and Action each has a relationship type, "Related_to" with the Super_Info_Object.
- ECA_Rule has a relationship type, "Contain," with Event, Condition, and Action (ECA) entity types.

It is important to note that:
(1) There is one "aggregation" relationship type: contain.
An ECA_Rule Entity "contains" Event, Condition, and Action entities.
(2) There is one "generalization" relationship type: consist_of.
A Super_Info_Object entity type "consists_of" the following entity subtypes:
Database, Table, and Column entity types.

(3) There are two weak entity types: Table and Column.
The Table entity has an ID-dependency on a Database entity, and the Column entity has an ID-dependency on a Table entity.

This conceptual model also illustrates the **linkage** between the ECA_Rules and the database contents (such as databases, tables, and columns). Note that Figure 4.2-4 and Figure 4.2-5 are related: the "Table on entities" and "Table on relationships" in Figure 4.2-4 are merged into "table" entity type in Figure 4.2-5 (in other words, "table" entity type is a super type of the other two entity types).

4.3 Integration with Reverse Engineering

Each data source can be and should be reverse-engineered into a conceptual model. Then, these conceptual models can be integrated into a "unified" conceptual model. This "unified conceptual model" will be linked to the metadata model so that a change in one may trigger the changes in the other.

4.4 Integration with Active Modeling

Active conceptual modeling capability: This capability is a continual process of describing the relevant aspects of the domain including the activities and changes under different perspectives. At any given time, the conceptual model is viewed as a multi-level and multi-perspective abstraction of the domain representing the user's knowledge. Dynamic reverse modeling and change management techniques will be used to capture the constraint knowledge stored in the individual database system schema and its changes. Constraint management for resolving conflicts among data values from multiple sources will be applied to achieve global data consistency. The active conceptual model can enhance the interactive and dynamic modification of domain knowledge (e.g. confidence value of data), rules, and the ability to add and remove information in a flexible and consistent manner.

As can be seen from the metadata model in Section 4.2, The ECA (Event Condition Action) rules [Chak95] have been captured as entity types and relationships types in the metadata model. By doing so, we are on the way toward an integrated modeling and execution system.

5   A Decision-Support System for Information Conflict Resolution

A decision-support system will be very useful to the persons who need to make decisions based on conflicting data, each with a (possibly) different InfV value. A prototype has been implemented using Java for testing the results on two Oracle databases located in two nodes of a network (one served as a server and the other served as a client). The development language used is Java™ 1.2.2. The operating system used on the server is Windows NT™ 4.0. The database management system used is Oracle 8i.
We use Microsoft ODBC Administrator and Net8 Configuration Assistant to set up the oracle client services. We have Oracle 8i client installed on the client workstation running the Java application.

One of the screens of the software prototype is shown in Figure 5-1. The user can query the data, and the system displays the data from two databases simultaneously. The user can then asks the system's assistance in selecting one of the algorithms for making decision.

Let us explain the prototype system in more detail. There are two test databases located in two nodes in a local area network. The server station is running under Windows NT, and the client station is running under Windows 98. The test databases are implemented using Oracle 8i. The database schema is shown in the Appendix; it contains some data fields that may be needed in a military application. Both databases have the same schema, but the data are not completely the same. For example, the employee age in one database could be "34" while the age of the same employee in the other database could be "36".

The user can input SQL query into the blank "input area" at the top of the user screen. The system will respond and fill up the values in the blank areas in the middle of the screen and information validity (InfV) values extracted from both test databases. For example, from Test1 database, the "age" value of a particular employee could be "34" and the InfV value could be "0.8", while from Test2 database the "age" value of the same employee could be "36" and the InfV value could be "0.6". The user has a choice of algorithms listed in the left side of the screen. If the user needs explanation of a particular algorithm, the user can click on the buttons on the right-hand side for "explanations."

After the user clicks (chooses) a particular algorithm, the system will calculate the composite data value and the composite InfV value based on the selected algorithm and display the composite values in the blank areas in the middle of the screen. After the user satisfies with the new composite values, the user can confirm the choice by selecting the correct buttons in the lower right corner of the user screen. The user also has choices of picking one particular data value from one of the two databases. In addition, the user has the choice of overwriting both database values and all algorithms by inputting the data value he/she thinks is correct.

Figure 5-1. A User Screen of the Information Validity Assessment Decision-support System Prototype

Figure 5-2. A Screen Explaining the qualitative terms (such as high, medium, and low).

It is possible to use qualitative terms (such as "high," "medium," and "low,") to express InfV values instead of numeric values. However, in order to calculate the composite value of InfV value of the composite data value, we need to translate the qualitative value into numeric values. In Table 3 of the test databases (see Appendix), we keep track of the numeric equivalent values (actually, the range of values and the likely values) of each qualitative terms. When the user asks for explanation of qualitative terms, the system will display the explanation in a pop-up screen (Figure 5-2) so that the user can get an idea of what the qualitative term means.

6 Summary, Future Directions, and Relevance to Critical Infrastructure Protection and Information Assurance

6.1 Summary

We have outlined some of the important steps in a framework for Information Validity Assessment. We have also described some algorithms for conflict resolution. We have proposed a preliminary framework for information Validity assessment based on meta-data conceptual modeling methodology. We have developed the following concepts and computation formulae:

- Developed a framework and meta-data conceptual modeling methodology based on two-level ER Modeling technique.

- Developed the derivation of Information Validity Value of data items (using a weighted average type of formulae) based on three factors: (a) reliability of the hardware and software that the database resides, (2) freshness of the data, and (3) believability of the data.

- Developed a set of computational formulae or algorithms for conflict resolution: We have developed the computation formulae for conflict resolution between two data sources or for many data sources (i.e, greater than 2 data sources). We have also developed computation formula for the information validity value of the whole database (or the whole table) from the information validity value of each data item.

- Developed a meta-data model for modeling the relationship between information validity and active database information.

In addition, we have implemented a prototype of a decision support system using Java and Oralce DBMS version 8i for helping people to make decisions under conflicting data situations.

6.2 Future Research Directions

Further research work can be done in the near future. For example, we may consider these extensions: adding a meta-data model, incorporating active modeling, and applying reverse engineering concepts and techniques into the framework. In meta-data modeling and active data modeling, we may consider the use of the Entity-Relationship (ER) model [Chen76] or one of its extensions. In terms of applications of the framework and techniques mentioned in this paper, one of such application is to assist the identification of the culprits during or after the information attacks in cyberspace [Chen97, Chen98a]. A military application is to incorporate the techniques in the architecture, design, and implementation of Joint Battle Space proposed by the Air Force Science Advisory Board [SAB99, Chen00c]. In addition, the software prototype can be extended to accommodate 3 databases or more.

Research Direction #1: To investigate techniques of how to get the information validity values of each data elements (or columns, tables, databases) in practice and how to do the computation when these values are missing or unavailable. The computational formulae we developed in the previous project rely on the values of information validity that are known and available. How can we get these values in practice? If we cannot get these values from actual data collected on the databases or systems in the past, how can we get the best estimates from the so-called, "experts". If we have more than one expert, which one should we trust more? And how?

Also, is "relying experts" the only proposed approach? The answer is "no." There is a need to investigate how to get the "estimates" if "experts" are not available.   Another type of research problems is: do the current algorithms assuming confidence values exist for all data elements? What if there is none or only a few exists? In that case, should we extrapolate the unknown values from the known values or rely on other techniques?   Another type of research problems is the granularity of the validity of data. What is the significance of getting the validity of a data element, a column, a row, a table, a database, or a collection of databases? What is the optimal level of granularity?

Research Direction #2: To refine the existing meta-data model: We have developed a meta-data model in the previous project. It will be useful to check with some real world data and database systems to see whether the meta-data model is robust or not. There is a need to find out whether any important features (entity types, relationship types, and attributes) are missing in the existing meta-data model. If so, we need to add those features to the meta-data model.

Research Direction #3: To investigate how to automatically trigger the checking of data consistency and the computation of the best-estimated data values. We plan to study how to link the meta-data databases with production databases and information validity assessment computation formulae so that the data consistency checking can be triggered automatically or at least semi-automatically. It will be useful to study how the computation formulae can be activated and fed with the accurate meta-data needed in the computation.

Research Direction #4: To investigate whether XML-based techniques will be directly relevant to our approach: We plan to study whether several XML-based techniques (such as XPointers, XLinks, RDF, and DAML) will be useful and directly relevant to our approach. We will focus on whether we can incorporate one or more of these techniques "immediately" to be useful to our project.   We are interested in something that can be implemented and demonstrated within the project performance period and not interested in the benefits that needs extensive research and developed efforts.

Research Direction #5: To investigate how to characterize data pedigree and relate it to information validity assessment: Data pedigree is an important source of meta-data. We plan to investigate how to characterize this kind of meta-data and what is the best way to input and to store this kind of meta-data with respect to our approach. There are some COTS software available for input and display data pedigree charts, primarily for genetic data (for example, the PROGENY4 software package offered by Progeny Software LLC). It

will be useful to study these COTS software and to see whether we can adopt them into our approach.

Research Direction #6: To refine the reverse engineering methodology and tools to build the conceptual model from existing database schemas: Many existing databases were developed without a conceptual data model. It will be very useful to derive the conceptual data models by reverse engineering the existing relational database schemas. In the previous project, we have performed a preliminary investigation on the possible use of these reverse-engineering COTS tools (PowerDesigner by Sybase and ERWIN by Computer Associates). It will be interesting to study the imported and exported data formats of these tools to see how they can be integrated into our software prototype.

Research Direction #7: To implement a meta-data database and to refine the software prototype of the decision support system: We have developed a simple user interface as a way to demonstrate how a decision system might help operational personnel and decision makers in resolving conflicting data from different data sources. It will be useful to implement a better user-friendly interface with additional features to demonstrate the feasibility and utility of the project. In addition, it may be desirable to implement a meta-data database using Sybase or Oracle and to populate the database with hypothetical/sensitized meta-data for demonstration purposes.

Research Direction #8: Are the credibility assessment and conflict resolution techniques different from the file environment to the relational database environment, and from numeric to non-numeric data? In other words, the framework should consider heterogeneous data environment, not just different types of DBMS's but also different types of data.

6.3   Relevance to Critical Information Protection and Information Assurance (CIPIA)

This research is very relevant to Critical Information Protection and Information Assurance (CIPIA) in the following ways:
- A lot of intelligence information comes from sources with various degrees of credibility level. How to assess and keep track of the credibility of this information is a crucial problem.
- It is common that the intelligence data for the same event/item may be conflicting with each other. How to assess the "real value" (or the best estimate" under the circumstances and available information) is another critical problem the operational personnel and the analysts are facing everyday.
- One of the most critical problems the U.S. is facing today is not enough information but rather how to integrate the information available and how to make intelligent use of the information.

Our current research and future research along the similar directions will be useful to solve these CIPIA problems

References

[ArFr99]   Artale, A. and E. Franconi, ”Reasoning with Enhanced Temporal Entity-Relationship Models,” <u>Proc. of the International Workshop on Spatio-Temporal Data Models and Languages</u>, Florence, Italy, IEEE Computer Society Press, 1999..

[CDLN99]   Calvanese, D., G. De Giacomo, M. Lenzerini, D. Nardi, and R. Rosati , “A Principled Approach to Data Integration and Reconciliation in Data Warehousing,” <u>Proc. of the International Workshop on Design and Management of Data Warehousing (DMDW’99)</u>, Heidelberg, Germany, 1999..

 [Chak95]   Chakravarthy S., Z. Tamizuddin and J. Zhou , “AVisualization and Explanation Tool for Debugging ECA Rules in Active Database,” Proc. of the 2nd International Workshop on Rules in Database Systems (RIDS 95), Athens, Greece, October 1995. Springer-Verlag, 1995, pp. 221-233.

[Chen76]   Chen, P. P. "Entity-Relationship Model: Toward a Unified View of Data," ACM Transactions on Database Systems, Vol. 1, No. 1 (March 1976).

[Chen 96]   Chen, P. P., "Efficient Data Retrieval and Manipulation using Boolean Entity Lattice," (with A. Yang), <u>Data & Knowledge Engineering</u>, <u>Vol. 20</u>, 1996, pp.211-226.

[Chen97]   Chen, P. P., *Reconstructing the Information Warfare Attack Scenario: Guessing What Actually Had Happened Based on Available Evidence*, Research Report, Air Force Research Laboratory, Information Directorate, Rome, NY, September 1997.

[Chen98a]   Chen, P. P., *Reconstructing the Information Warfare Attack Scenario*, Proceedings of Infowar Symposium, Washington, D.C., May 1998 (slides only).

[Chen98b]   Chen, P.P., Akoka, J. , Kangassolo, H., and Thalheim, B. (eds.), <u>Conceptual Modeling: Current Issues and Future Directions</u>, Springer-Verlag, Berlin, Lecturing Notes in Computer Sciences, No. 1565, 1998.

[Chen98c]   Chen, P.P. Thalheim, B., and Wong, L. “Future Directions of Conceptual Modeling,” in: Chen, P.P., Akoka, J. , Kangassolo, H., and Thalheim, B. (eds.), <u>Conceptual Modeling: Current Issues and Future Directions</u>, Springer-Verlag, Berlin, Lecturing Notes in Computer Sciences, No. 1565, 1998.

[Chen99]   Chen, P. P., Embley, D.W., Kouloumdjian, J., Liddle, S.W., Roddick, J.F. (eds.), <u>Advances in Conceptual Modeling</u>, Springer-Verlag, Berlin, Lecturing Notes in Computer Sciences, No. 1727, 1999.

[Chen00a]   Chen, P. P., “The Importance & Major Research Issues In Intelligent Knowledge Based Cyber Forensics For JBI,” Research Report, Air Force Research Laboratories, Information Directorate, Rome, NY, September 2000.

[Chen00b]   Chen, P. P., "A Survey of Forensics Programs in U.S. Colleges," A report submitted to Air Force Research Laboratories, Information Directorate, Rome, NY, September 2000.

[Chen00c]   Chen, P. P., *Application of Information Validity Assessment Techniques in JBI,"* Joint Battle Space (JBI) Invited Workshop organized by AFRL, Minnowbrook Conference Center, NY. October 2000, (presentation only).

[Chen01a]   Chen, P.P., Project Status Report, Submitted to SPAWAR SYSCEN, February 2001.

[Chen01b]   Chen, P.P., "Information Validity Assessment in Integrating Heterogeneous Data Sources," Proc. of 4th International Conference on Information Fusion, Montreal, Canada, August 7-10, 2001.

[Chen02]   "Entity-Relationship Modeling: Historical Events, Future Trends, and Lessons Learned," in: Software Pioneers: Contributions to Software Engineering, Broy, M., and Denert, E. (eds.), Springer-Verlag, 2002.

[Demp76]   Dempster, A., "Upper and lower probabilities induced by a multivalued mapping," *Annals of Mathematical Statistics 38*: 325-339, (1976).

[Hsu   ]   Hsu, C., et. al., "The Metadatabase Approach to Integrating and Managing Manufacturing Information," Journal of Intelligent Manufacturing, forthcoming.

[IEEE99]   Proceedings of Third IEEE Workshop on Meta-Data, Bethesda, MD, April 6-9, 1999, IEEE Computer Society.

[JaVa97]   Jarke, M. and Y. Vassiliou, "Data Warehouse Quality Design: A Review of the DWQ Project," *Proc. 2nd Conference on Information Quality*. Massachusetts Institute of Technology, Cambridge, 1997.

[JeQJ98]   Jeusfeld, M.A., Quix, C., and M. Jarke, "Design and Analysis of Quality Information for Data Warehouses," *Proc. 17th International Conference on Conceptual Modeling (ER'98)*, Singapore, Nov 16-19, 1998.

[SAB99]   Scientific Advisory Board, US Air Force, *Report on Building the Joint Battlespace Infosphere, Vol. 1: Summary*, SAB-TR-99-02, December 17, 1999.

[Shaf76]   Shafer, G. *A Mathematical Theory of Evidence,* Princeton University Press, Princeton, New Jersey, 1976.

[StJa96]   Staudt, M. and M. Jarke, "Incremental Maintenance of External Materialized Views," Proceedings of 22nd VLDB Conference, Mumbai, India, 1996.

[StWa98]   Storey, V. and R. Wang, "Modeling Quality Requirements in Conceptual Data Base Design," Proceedings of 1998 Conference on Information Quality, pp. 64-87.

APPENDIX A
Testing Database

The testing databases have several tables. One of them (Table 1) looks like the following:

```
EVENT_TYPE          NUMBER(3),
 TIME          DATE,
 TRACK_NUMBER  NUMBER(10)   NOT NULL,
 PLATFORM   NUMBER(3),
 COURSE       NUMBER(5),
 SPEED             NUMBER(5),
 ALT_DEPTH   NUMBER(15),
 RANGE             NUMBER(10),
 BEARING      NUMBER(10),
 CALL          NUMBER(10),
 SIGN          NUMBER(10),
 MODE_1            NUMBER(10),
 MODE_2            NUMBER(10),
 MODE_3            NUMBER(10),
 MISSION           NUMBER(10),
 NATIONALITY       CHAR(20),
 SOURCE            NUMBER(10),
 ID_SOURCE   NUMBER(10),
 TRACK_TYPE        NUMBER(10),
 ID            NUMBER(10) NOT NULL,
 CATEGORY    NUMBER(10),
 SYMBOLOGY         NUMBER(10),
 LAT_N_S      CHAR(2),
 LAT_DEGREE        NUMBER(2),
 LAT_MINUTE        NUMBER(2),
 LAT_SECOND        NUMBER(10),
 LON_E_W      CHAR(2),
 LON_DEGREE        NUMBER(2),
 LON_MINUTENUMBER(2),
 LON_SECOND        NUMBER(10)
```

The 2nd table contains the meta data. This table describes the InfV values of each column of a particular table (for example, Table 1):

```
TBL_NAME    VARCHAR2(30),
COL_NAME    VARCHAR2(30),
VALIDITY     VARCHAR2(20),
PRIMARY KEY (TBL_NAME, COL_NAME)
```

The third table (Table #3) contains the interpretation of some qualitative terms such as "high," "medium," and "low". The schema of Table is shown below:

```
QUALITATIVE-TERM        VARCHAR2(10),
PERCENT_LOW       NUMBER(4),
PERCENT_HIGH       NUMBER(4),
PERCENT_ESTIMATION     NUMBER(4),
PRIMARY KEY (QUALITATIVE-TERM)
```

Evaluation of Two Commercial Reverse-Engineering Tools

We have studied the information provided by the vendors of two popular commercially available reverse-engineering tools: PowerDesigner and ERWIN. The functionality of these two tools are summarized in the following few pages.

Based on the requirements of our project, we think PowerDesigner will fit our needs better than ERWIN.

Reverse Engineering feature of the PowerDesigner 7.5

Reverse engineering a database schema (PDM stands for Physical Data Model)
Reverse engineering is the process of generating a PDM, or specific PDM objects, from an existing database schema.

1.   There are two ways to generate a PDM from a database schema:
Generate PDM using
1. Script file
You reverse engineer an SQL script which contains creation statements. This is normally the script used to generate the database

2. ODBC data source
You reverse engineer the schema for an existing database, specifying an ODBC data source, and connection information

Generating a PDM from a database
When you reverse engineer a database schema using an ODBC data source, you can choose to generate a PDM for all objects, or selected objects, in the database.
The object types that you can reverse engineer are DBMS-dependant. Unavailable object types do not appear for selection.

2. Object types for reverse engineering
Using an ODBC data source, you can select the following object types for reverse engineering:

Tables
Views
System tables
Synonyms
Users
Domains
Triggers
Procedures
Tablespaces and storages

Reverse engineering users
  Only users that have creation rights are reversed engineered.

3. User-defined and abstract data types

You can reverse engineer user defined and abstract data types. In the generated PDM, the names of these data types appear in the List of Abstract Data Types.

4. Reverse engineering choices

You can reverse engineer either into a new or into an existing PDM. Reverse engineering into an existing PDM involves a merging of both sources into an updated PDM.

5. Filters and options for reverse engineering

You can use filters to restrict the number of objects to reverse engineer.
Certain object types have attributes, or options, that you can select to be included in the generated PDM.

a. Filters

You can restrict database objects to reverse engineer by selecting an owner or a database qualifier. The following filters are available:

1. Qualifier
A qualifier is a database or partition in a database that contains one or more tables. When a qualifier is selected as a filter, it restricts the objects available for reverse engineering to the objects contained within the selected qualifier.
For example, the DB2 DBMS authorizes the use of the qualifier field to select which databases are to be reverse engineered from a list.

2. Owner
Normally the creator of a database object is its owner. When Owner is selected as a filter; it restricts the objects available for reverse engineering to the objects owned by the selected owner.

Note: Selecting objects from multiple owners
To reverse engineer objects from multiple owners, you can select All as a filter from the Owner dropdown listbox. All the objects belonging to all owners appear in the list, and you can select the objects for reverse engineering regardless of their owner.

b. Reverse options

Reverse engineering options are dependant on object type, and the selected DBMS. To display, or modify, the reverse engineering options for an object type, you click the appropriate page tab in the ODBC Reverse engineering dialog box. Unavailable options appear grayed.
You can select reverse engineering options for the following object types:

Object type	Option type
1. Table	Indexes
	Checks
	Physical options
	Primary keys
	Foreign keys
	Alternate keys
2. View	Reverse as table

2. Reverse engineering into a new PDM

You can reverse engineer an existing database into a new PDM. The data source can be either from a script file or an ODBC data source.

a. Reverse engineering from a script file
b. Reverse engineering from an ODBC data source
c. Reverse engineering into an existing PDM

You can also reverse engineer into an existing PDM.

ii. Merging two PDM

When you reverse engineer into an existing PDM, a model merge window appears after the reverse engineering process is complete. You can then use the model merge function to integrate the reversed objects into a current model.

iii. Automatic archiving

When you merge database objects that have been reverse engineered into a current PDM, you can choose to archive the newly generated PDM, by selecting the Automatic Archive checkbox from the model merge window.

For more information on comparing and merging two models, see the chapter Comparing and Merging Models in the PowerDesigner General Features Guide.

a. Reverse engineering a Microsoft Access 97 database

PowerDesigner and MS Access 97 use .DAT files to exchange information. These files are reversed into the PDM. The access.mdb database uses or creates .DAT files to reverse Access databases. You can define the database reverse parameters from the access.mdb database window.

b. Generating a PDM from a database creation script

You can generate a PDM, or add PDM objects directly from a database creation script.

2. ERWIN Functionality

Reverse Engineering Capabilities. Existing data assets and expertise can speed the delivery of new systems and improve their overall quality. ERWIN let the users reverse-engineer existing systems and incorporate these designs as part of the new development effort. It is possible to create template models containing reusable design components and apply them to new models, jump-starting the data design process.

Extensive Platform Support. Because IT organizations often rely on many different database platforms, ERWIN supports a wide selection of server and desktop databases. ERWIN is tuned to work with each of these database environments, enabling the optimization of the database design and the optimization of the performance of your database. ERWIN models can be used to generate the same design to multiple platforms, or to convert an application from one database platform to another.

Supported Environments
Databases:
• Ingres ® II • Oracle
• CA-Clipper ®
• Paradox
• DB2 • Rdb
• dBASE
• Red Brick Warehouse
• FoxPro
• SAS
• HiRDB
• SQL Anywhere
• INFORMIX
• SQLBase
• InterBase
• Sybase
• Microsoft Access
• Teradata
• Microsoft SQL Server
• ODBC 2.0, 3.0
Platforms:
• Windows 95, 98, NT 4.0